

## **REMARKS**

Applicants amend independent claims 1, 8, and 15, and dependent claims 3 and 17. Further, Applicants cancel claims 2, 10, and 16. Claims 1, 3-9, 11-15, and 17-20 remain pending. Independent claims 1, 8, and 15 have been amended to include subject matter from the dependent claims. Accordingly, Applicants respectfully submit that the amended claims are entitled to entry after final. No new matter is added by way of the amended claims, which are fully supported by the specification.

### **Claim Rejections Under 35 U.S.C. § 102(b)**

The Examiner rejected claims 1 and 15 under 35 U.S.C. § 102(b) as being anticipated by Herman et al. (U.S. Patent No. 5,954,826) in paper #3. Further, the Examiner rejected claim 8 as being anticipated by Mehring (U.S. Patent No. 5,675,729) and thereafter, rejected claims 8 and 12 as being anticipated by DeLong et al. (U.S. Patent No. 5,892,947). Applicants respectfully traverse.

The amended claims, made for clarification, recite features not disclosed by individual references to Herman et al., Mehring, or DeLong et al. The references do not disclose *a get assertion class that obtains assertions from a specification*. Further, among other feature differences, Herman et al. does not teach providing a plurality of classes such that *each class is capable of performing a particular task* related to obtaining information from a specification. Additionally, Mehring does not teach a *computer program for obtaining assertions from a specification for a computer program* such that *a code segment parses the identified context to obtain assertions*. Similarly, DeLong et al. does not disclose the analysis of a *context to obtain assertions*. DeLong et al. also does not teach the *parsing of a context to obtain assertions*.

Thus, Applicants respectfully submit that independent claims 1, 8, and 15 are not anticipated by Herman et al., Mehring, or DeLong et al. Accordingly, Applicants respectfully request the withdrawal of the 35 U.S.C. § 102(b) rejections.

### **Claim Rejections Under 35 U.S.C. § 103(a)**

The Examiner rejected claims 2 and 16 under 35 U.S.C. § 103(a) as being unpatentable over Herman et al. in view of Buxton et al. (U.S. Patent Pub. No. US 2002/0089526). Further, the Examiner rejected claims 3 and 4 as being unpatentable over Herman et al. and Buxton et al. in view of Golshani et al. (U.S. Patent No. 5,850,631). Next,

the Examiner rejected claims 5-7 and 18-20 as being unpatentable over Herman et al. in view of Tan (U.S. Patent Pub. No. US 2002/0138510). Thereafter, the Examiner rejected claims 9, 10, 13, and 14 as being unpatentable over DeLong et al. in view of Gramlich (U.S. Patent No. 5,826,025) and rejected claim 11 as being unpatentable over DeLong et al. and Gramlich in view of Schaffer (U.S. Patent No. 5,911,041). Lastly, the Examiner rejected claim 17 as being unpatentable over Herman et al. and Buxton et al. in view of Golshani et al. Applicants respectfully traverse.

Herman et al. and Buxton et al.

Herman et al. discloses a graphical user interface (GUI) for selecting a specification file and class files. The reference further discloses that the class files are Java program files. The GUI permits the selection of class files for comparison with the specification file. For example, the reference discloses on Figures 1 to 3 that specification and class files can be added within the GUI. Thereafter, Figures 4-6 illustrate that the class files are analyzed, and further discuss how the class files do not conform to the specification files. Thus, the program/class files are selected to test for compatibility with the specification files. During the analysis of the specification and class files, four types of errors can be displayed. These errors are part of the analysis, which determines the compatibility of the specification and class files. For example, one type of error is "a resource or dependency that is required by the program file that is lacking in the specification." Column 5, lines 21-25.

Buxton et al. discloses a mechanism to add GUIs in a Java environment. Specifically, the GUI, called InfoCenter, is communicatively coupled to multiple applications. The applications can be Java applets, beans, or components. When an application announces that it needs a GUI, the InfoCenter provides a user interface for the application. The user interface is established by the application announcing selection data. The selection is wrapped up in a SelectionContext object and sent to the InfoCenter via an InfoBus. Further, a SelectionContext class allows the InfoCenter to get parameters for a selection assertion, which is stored in an instance of this class.

Thus, the references disclose how to determine the compatibility of specification and class files, and how to produce user interfaces in response to announcements from applications.

In contrast, independent claims 1 and 15 recite collecting information on a specification by providing a plurality of classes that can obtain information from the

specification. One type of information obtained by a class, such as a get assertion class, is an assertion. A command requests that a task be performed and a class is selected from the plurality of classes based on the requested task. Further, the selected class runs, such that information on the specification is obtained.

As compared to the claimed invention, the references do not disclose a method of obtaining assertions of a specification. Further, the classes of Herman et al. do not obtain assertions from the specification. Instead, the reference simply determines whether the classes and specification are compatible. Further, Herman et al.'s specification and classes do not show how after receiving a command, the command requests the performance of a task that runs a selected class to obtain information from the specification. Instead of running a class to obtain information, Herman et al. compares class files and specifications for error analysis, such as determining dependencies between the files. Thus, Herman et al. has no need to examine specifications for assertions because classes are simply tested for compatibility with specifications.

The Office offers Buxton et al. for the proposition that the plurality of classes includes a get assertion class that obtains assertions from the specification. However, checking for compatibility errors in Herman et al. and getting parameters for a selection assertion in Buxton et al. do not combine to provide motivation to *obtain assertions from a specification*. The assertion of Buxton et al. is a selection assertion stored in an instance of the SelectionContext class. The SelectionContext class is a helper function that is the only connection mechanism between the application and the InfoCenter to announce selection information on the InfoBus. The selection information is used to generate the user interface. Thus, the selection assertions of Buxton et al. have no purpose in Herman et al. because Herman et al. simply illustrates how to test the compatibility of classes and specifications. Herman et al. also does not require the acquisition of assertions to communicate between an application and a GUI, as disclosed by Buxton et al.

Buxton et al. also does not disclose how to examine specifications for assertions. Specifically, Buxton et al. does not disclose how an assertion should be obtained from a specification. When combining the assertions of Buxton et al. with the specifications of Herman et al., the result does not obtain assertions from the specification, as recited by the independent claims. Thus, the functional characteristics of the inventions disclosed by the references do not render obvious the claimed invention.

Applicants respectfully submit that the references, singly or in combination do not teach or suggest the claimed invention recited by independent claims 1 and 15. Further, the references do not render obvious the dependent claims 3-7 and 17-20, which depend from independent claims 1 and 15, respectively, for at least the same reasons. Thus, Applicants respectfully request the withdrawal of the rejection under 35 U.S.C. § 103(a).

DeLong et al.

DeLong et al. is offered to render obvious independent claim 8. However, the reference discloses how to automate software test generation and synthesis based on a selected software specification. In column 5, lines 44-57, the reference discloses preconditions and postconditions of a test case specification. Further, in Figure 10, the reference discloses an overall method of the invention. The overall method does not disclose features of receiving an input specification such that a context is identified within the input specification. Further, the overall method does not illustrate how to parse the identified context to obtain assertions such that the assertions are testable statements within the input specification. Finally, the overall method does not illustrate how to add the obtained assertions to an assertion result set such that the result set can be used to facilitate testing of the input specification.

Similar to Figure 10, DeLong et al.'s Figure 2 also does not disclose the recited features of independent claim 8. Specifically, the teaching of a logical language processor to logically describe selected software to produce a test case does not render obvious obtaining assertions from a specification. The assertions are added to a result set used to facilitate testing of the specification, which is not taught or suggested by the reference.

Thus, Applicants respectfully submit that independent claim 8 and dependent claims 9 and 11-14 are allowable and the withdrawal of the rejection under 35 U.S.C. § 103(a).

Application No. 09/881,822.  
Final Office Action mailed 3/31/04.  
Response to Final Office Action mailed 6/1/04.

Applicants respectfully request a Notice of Allowance based on the foregoing remarks. If the Examiner has any questions concerning the present amendment, the Examiner is kindly requested to contact the undersigned at (408) 749-6900. If any other fees are due in connection with filing this amendment, the Commissioner is also authorized to charge Deposit Account No. 50-0805 (Order No. SUNMP013). A copy of the transmittal is enclosed for this purpose.

Respectfully submitted,  
MARTINE & PENILLA, LLP

A handwritten signature in black ink, appearing to read 'Feb Cabrasawan', followed by a long horizontal line extending to the right.

Feb Cabrasawan  
Reg. No. 51,521

Martine & Penilla, LLP  
710 Lakeway Drive, Suite 170  
Sunnyvale, California 94086  
Tel: (408) 749-6900  
Customer Number 32291